Microtuning of Membership Functions: Accuracy vs. Interpretability

Qiang Shen and Javier G. Marín-Blázquez
{qiangs,javierg}@dai.ed.ac.uk
Centre for Intelligent Systems and their Applications
Division of Informatics, The University of Edinburgh

ABSTRACT. *A major disadvantage of existing methods for tuning descriptive fuzzy models is that the usual constrains over the changes on the fuzzy membership functions do not guarantee that no radical changes in the definitions and hence, no unacceptable disruptions in the interpretability of the original model would take place. This paper proposes a new tuning method, called* microtuning, *which avoids drastic changes by enforcing that the possible loss in interpretability is kept to minimal. This is achieved by ensuring the modified sets to have, at least, a given degree of similarity with their original. The paper focuses on the issue of how accuracy increases as the similarity constraint is relaxed. It reveals the tradeoff between losing interpretability and gaining precision in tuning a descriptive model. Simulation results show that most of the improvement in model accuracy can be obtained without major changes in the original set definitions, microtuning may be all what is required.*

I. Introduction

Computing with words is a fundamental contribution of fuzzy logic [13]. This is feasible via the utilisation of linguistic variables which are variables whose values can be words rather than numbers. These words can be interpreted as semantic labels to the fuzzy sets employed within the fuzzy models [12]. Thus, human comprehensible computer representation of domain problems can be created when desired.

However, a great majority of fuzzy sets used in fuzzy models are created and tuned to best fit the data available. They are not encoded to keep the meaning of the semantic labels, following the so-called *approximative approach*, or precise fuzzy modelling. (Note that the word *approximative* is herein used instead of approximate to mirror the word *descriptive* in descriptive modelling which is itself an approximate approach.) They tend to create fuzzy sets that fit the data very well but that usually lack features which are considered important to make it easy for human users to interpret the resulting model and its reasoning [10].

Opposing this stands the *descriptive approach*, or linguistic fuzzy modelling. Here, semantics are as important as accuracy. The definition of the fuzzy sets is human given. For pure descriptive modelling no changes to such definition are allowed [1], [2], [11]. However, this harsh constraint leads to coarse partitions of the underlying value ranges of the linguistic variables and hence significantly limits the accuracy of the model to be built.

To improve model flexibility, the *pseudo-descriptive* approaches [5] try to regain interpretability by aggregating approximative fuzzy sets until some descriptive features are obtained. Others impose certain restrictions over the tuning of descriptive fuzzy sets by modifying the membership function definitions. However, once such definitions are allowed to change model transparency can not be guaranteed. Several modelling constrains have been proposed to avoid such potential loss of interpretability (e.g. keeping the relative order of the sets, and ensuring coverage and/or distinguishability) [10].

Unfortunately, such restrictions, even applied jointly, may lead to cases where the sets become so different to the original that the intended interpretative power is lost. In this paper a new pseudo-descriptive method is proposed. This method, called *microtuning* avoids drastic changes by enforcing that the possible loss in interpretability is kept to minimal by ensuring the modified sets to have, at least, a given degree of similarity with their original. The tuning is implemented by differential evolution [8], an evolutionary method proven to be very good at unconstrained membership function tuning [4]. The paper focuses on the issue of how accuracy increases as the similarity constraint is relaxed. It reveals the tradeoff between losing

0-7803-7280-8/02/$10.00 ©2002 IEEE

interpretability and gaining precision in tuning a descriptive model.

The rest of the paper is organised as follows. Section II gives an overview of the problem under investigation. Section III shows the induction algorithm used to create descriptive fuzzy rules (though other descriptive modelling methods may be used as alternative). Section IV explains the differential evolution algorithm adopted to tune the membership functions. Section V reports on typical simulation results, demonstrating the tradeoff between interpretability and accuracy. The paper is concluded in section VI.

II. The Problem

The task of descriptive modelling is to find a finite set of descriptive rules capable of reproducing the input-output behaviour of the system being modelled. For classification problems, without losing generality, the system is assumed to be MISO (Multi-Input Single-Output), namely, a system of M inputs and one output that can be described by a set of K rules such as:

$$\begin{array}{c}\text{If } x_1 \text{ is } D_i^1 \text{ and ... and } x_M \text{ is } D_i^M \\ \text{then } y \text{ is } Class_h\end{array} \quad (1)$$

where i indexes the number of a rule ($1 \leq i \leq K$), x_j is the jth input variable ($1 \leq j \leq M$), D_i^j is a descriptive fuzzy set for x_j, and y is the output variable to be assigned to one of the possible output classes. The descriptive fuzzy sets involved are human defined and fixed throughout the rule generation process, in order to yield readily human-comprehensible models.

The information about the behaviour of the system under consideration is assumed to be a set of N input-output example pairs, with N usually being a large number:

$$\{(x_{t1}, x_{t2}, \ldots, x_{tM}, y_t), t = 1, \ldots, N\} \quad (2)$$

The ruleset to be induced is required to approximate the function $\varphi : X^M \rightarrow Class\ Y$ (that theoretically underlies the system behaviour) in the most consistently possible way with the given examples. Here $X^M = (X_1 \times X_2 \times \ldots \times X_M)$, with $X_1, X_2, \ldots, X_M$ being the domains of the inputs, and Y is the domain of the output classes. The rule induction task is to create the smallest possible subset of fuzzy rules that characterises the dataset to a degree as high as possible.

After such a ruleset is found, it may be possible to improve its precision if some slight changes are made to the definitions of the fuzzy sets given. However, unconstrained modifications of the fuzzy sets may result in drastic changes that totally ruin the semantic interpretation that the original ruleset entails. The present work proposes to use a similarity constraint imposed over the membership function modification to minimise the potential of such disruption.

The similarity measure used is defined as:

$$I(F_1, F_2) = \frac{A(F_1 \cap F_2)}{(A(F_2) + A(F_1)) - A(F_1 \cap F_2)} \quad (3)$$

with $A(F)$ denoting the area of the fuzzy set F. Thus, if α is the minimal similarity to be enforced a *microtuning* will ensure that:

$$I(M_{D_i^j}, D_i^j) \leq \alpha \quad (4)$$

where $M_{D_i^j}$ denotes the modified (tuned) version of the descriptive set D_i^j.

This measure is rather strong; high similarity values are only obtainable when the two sets are almost the same. Examples of the similarity measures between a fuzzy set and its shifts and those between two different sets are shown in Figures 1 and 2, respectively.

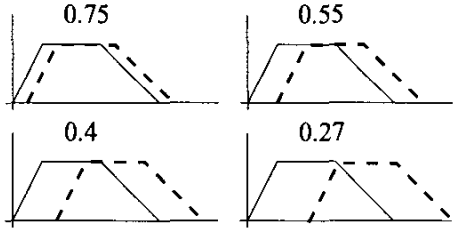

Fig. 1. Similarity values for the same set shifted

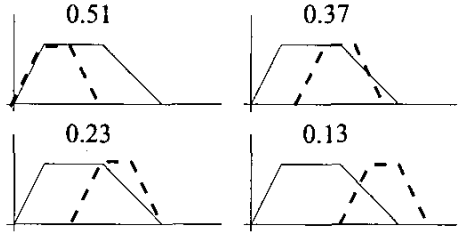

Fig. 2. Similarity values for different sets

III. Rule Induction

In this research, to obtain the fuzzy ruleset, Lozowski's pure descriptive induction algorithm [1] is

used. This algorithm works by exhaustive search, but the problems tested are small enough to allow this. For scaled-up applications of the ideas given here an alternative descriptive modelling method may be necessary.

Lozowski's algorithm generates a hyperplane of candidate fuzzy rules (see Equation 1) by fuzzifying the entire dataset using all permutations of the inputs. Thus, a system with M inputs, each of which has a domain fuzzified by f_j fuzzy sets ($1 \leq j \leq M$), the hyperplane is fuzzified into $\prod_{j=1}^{M} f_j$ M-dimensional clusters, each representing one vector of rule preconditions. Each cluster $\underline{p} = \langle D^1, D^2, \ldots, D^M \rangle$ may lead to a fuzzy rule, provided that dataset examples support it.

To obtain a measure of what classification applies to a cluster, fuzzy min-max composition is used. The input pattern of each example is fuzzified according to the fuzzy sets $\{\mu_{D^1}, \mu_{D^2}, \ldots, \mu_{D^M}\}$ that make up cluster $\underline{p}$. For each example $\underline{x} = \langle x_1, x_2, \ldots, x_M \rangle$, the t-norm of it with respect to cluster $\underline{p}$ and classification c is calculated as follows:

$$T_c^{\underline{p}}\underline{x} = \min\left(\mu_{D^1}(x_1), \mu_{D^2}(x_2), \ldots, \mu_{D^M}(x_M)\right) \tag{5}$$

To give a measure of the applicability of a classification to cluster $\underline{p}$, the maximum of all t-norms with respect to $\underline{p}$ and c is then calculated and this is dubbed an s-norm:

$$S_c^{\underline{p}} = \max\left\{T_c^{\underline{p}}\underline{x} \mid \underline{x} \in C_c\right\} \tag{6}$$

where C_c is the set of all examples that can be classified as c. This is iterated over all possible classifications to provide a full indication of how well each cluster applies to each classification.

A cluster generates at most one rule. The rule's preconditions are the cluster's M co-ordinate fuzzy sets. The conclusion is the classification attached to the cluster. Since there may be t-norms for more than one classification, it is necessary to decide on one classification for each of the clusters. Such contradictions are resolved by using the *uncertainty margin*, ε ($0 \leq \varepsilon < 1$). A t-norm assigns its classification on its cluster if and only if it is greater by at least ε than all other t-norms for that cluster. If this is not the case, the cluster is considered undecidable and no rule is generated. The uncertainty margin introduces a trade-off to the rule generation process between the size and the accuracy of the resulting ruleset. In general, the higher ε is, the less rules are generated, but classification error may increase. A fuller treatment of Lozowski's algorithm in use for descriptive modelling can be found in [1], [6].

IV. Differential Evolution

Evolutionary computation extends classical search techniques by imitating the idea of natural evolution. A "population" of set definitions compete, mate and recombine, with only the ones that "best" fit the data, thereby surviving in the next generation. *Differential evolution* is a new evolutionary heuristic approach [8]. A specific algorithmatic implementation of this approach for tuning fuzzy membership functions was introduced in [4]. To be self-contained, however, the following gives an overview of how this algorithm works for the present application.

A particular definition of the fuzzy sets used in the rules is encoded as a vector of real parameters. The vector is perturbed using differences of other vectors present in the current population (hence the name of this method). The perturbation is implemented by applying the so-called *move operator* on the vectors. The resulting vectors replace the original if they each lead to a definition of the sets that better fit the data available, adjudged by a *fitness* function. In this work, the fitness function is implemented by checking the model accuracy upon which the population is sorted. The sorting is done by taking preference those members that satisfy the similarity restriction imposed (between each pair of the modified and original fuzzy sets that are employed in the emerging ruleset, as measured by Equation 3).

The move operator in differential evolution can be generally represented by

$$V_{New} = Mix(V_{Present}, W, C) \tag{7}$$

where

$$W = V + F \times (V_{Difference}) \tag{8}$$

In this definition, $V_{Present}$ is the parent vector; V is another member of the population, either randomly chosen or being the current best (depending on what detailed move scheme is adopted [7]); $V_{Difference}$ is a vector formed by differences among some other vectors of the population, with actual form again depending upon the scheme used; F is a constant acting as a "learning" factor, used to control the aggressiveness of the method; V_{New} is the new vector that will replace $V_{Present}$ if it is of better quality; and Mix stands for a certain function (see below for a specific instantiation of it) which swaps between parts of $V_{Present}$ and W, subject to the swapping probability of C.

A. Scheme DE/Rand/1 (R1) – The Scheme Used

In this work, the following particular scheme is adopted, where three different random members V_1, V_2, and V_3 are taken from the present population, and the V_{New} vector is generated using:

$$W = V_1 + F \cdot (V_2 - V_3) \quad (9)$$

Note that no matter what scheme is used, the random members taken are always required to be different amongst themselves and from the parent. The process of generating V_{New} in the scheme DE/Rand/1 can be illustrated as shown in Figure 3.

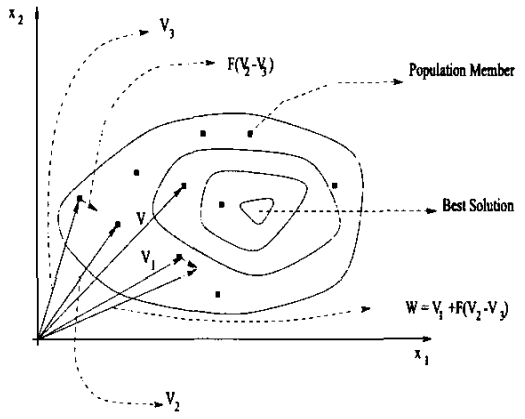

Fig. 3. Vector generation in Scheme DE/Rand/1

For computational simplicity, fuzzy sets used are assumed to be of a trapezoidal membership function. Such a fuzzy set can be represented by four parameters (see partition 2 of variable 1 in Figure 4 for example). The codification of the sets in each chromosome is illustrated in Figure 4. As can be seen the way they are encoded ensures that the partition of the variables' domains is strong: At most any underlying real point can belong to two different sets and the sum of the membership values of any point within the variable's domain is 1.

V. Simulation Results

A. Set-up

To demonstrate the proposed approach at work, typical classification problems are used here. The benchmark classification problems used are selected from [9], including the Breast Cancer, Diabetes, Iris, and Wine datasets. Table I summarises the set-ups of these datasets. Classification problems require the maximisation of the number of

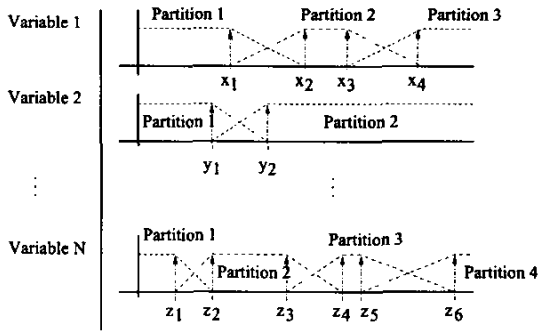

Fig. 4. Codification of membership functions in a chromosome

correctly classified samples whilst minimising the number of incorrectly or not covered cases.

TABLE I
CLASSIFICATION PROBLEMS

Name	Input	Classes	Samples	Rules
Breast C.	9	2	683	74
Diabetes	8	2	768	65
Iris	4	3	150	11
Wine	13	3	178	21

Fuzzification was carried out proportionally with respect to the size of the universe of discourse of the individual variables for each problem considered. That is, for each variable, the distance between its maximum and minimum value within the data set is divided such that all of them approximately cover an equal range of the underlying real values (with soft boundaries of course). The fuzzy sets resulting from such a partition are regarded as the given descriptive sets. This is not necessary in practical applications where this can, and should, be done by users or experts. For simplicity, each variable is allowed to take 3 linguistic labels.

In the present investigation, the parameter ε in Lozowski's algorithm is set up so as to obtain a sensible number of rules versus accuracy. This number (of rules) is also shown in Table I. Once rules are induced they are fixed and only the membership functions are modified. Note that although there exist tuning methods that may also modify rules this paper is focused on the tuning of membership functions only. Whilst tuning rules at the same time may improve model accuracy it may also blur the effect of microtuning the fuzzy sets, which is what is studied here. The goal of the

present experiments is not to obtain the most accurate model but to study how microtuning affects a given model.

To avoid possible overfitting each dataset has been separated into a training set containing 75% of all the given data and a test set comprising the remaining 25%. The similarity value α to be enforced was varied between 0 (unconstrained) and 1 (no modification allowed) in 0.05 increments. A hundred runs were executed for each problem and each similarity value. The best and mean values are shown in the results. Table II summarises the classification error percentage for the most interesting similarity values, where Trn and Tst stand for training and testing, respectively. In particular, the classification error rates of using the unmodified sets, which were used for pure descriptive fuzzy rule generation in the first place, are listed with respect to the similarity value of 1.

B. Results

Although the use of a particular similarity metric may have an impact upon the shape of the relation between similarity relaxation and model accuracy, a smooth and linear-like relation was expected. However, it has been very interesting to discover that this relationship is highly non linear. This result appears to be very similar for all problems tested, though it varies slightly as to where exactly a significant improvement of model accuracy starts and/or ends. As examples, Figures 5 and 6 show the simulation outcomes for the Breast Cancer and Wine problems. It can be seen that over the similarity value range from 0.65 to 0.85 the improvement is most significant. In particular, the shape of the curves is very sharp between 0.7 and 0.8. These values reflect quite high similarities between the modified and original fuzzy sets. This supports the idea of using microtuning, instead of free or weakly constrained tuning. Furthermore, model accuracy does not get any further significant improvement when the similarity value drops down to less than 0.6.

Lozowski's rule induction algorithm usually produces a fuzzy model that involves a high number of rules in order to obtain a good accuracy. This is, of course, expected from the use of any pure descriptive algorithm [2], [11]. Nevertheless, even though the number of rules has an impact upon the model accuracy it is not expected to affect dramatically this experimentally identified non-linear relation between accuracy and interpretability.

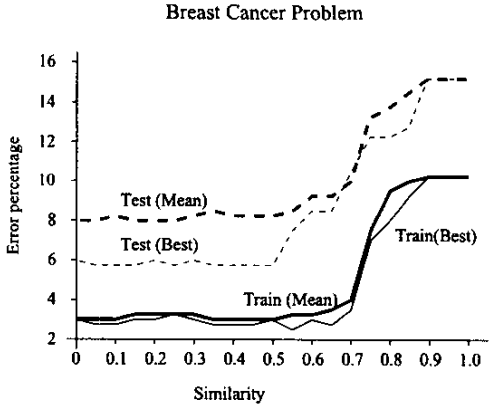

Fig. 5. Breast Cancer Problem

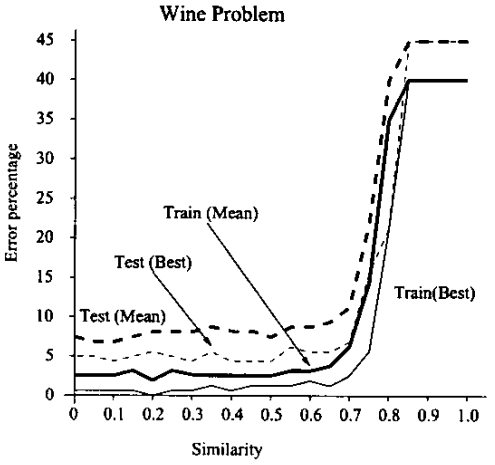

Fig. 6. Wine Problem

VI. Conclusions

A major disadvantage of existing methods for tuning descriptive fuzzy models is that the usual constrains over the changes on the fuzzy membership functions do not guarantee that no radical changes in the original set definitions and, therefore, no unacceptable disruption in model interpretability would take place. However, by introducing a constraint over the similarity between the original and the modified sets it is ensured that the possible loss in model interpretability can be kept to minimal. This is because the similarity value represents the relaxation degree allowed in the shape of fuzzy set definitions, so that no difficulties would arise in interpreting the semantics conveyed by the fuzzy sets. Thus, the modified and the original still refer to a similar concept. This is confirmed by systematic simulation results over

TABLE II

MODEL ACCURACY VS. SIMILARITY BETWEEN MODIFIED AND ORIGINAL FUZZY SETS

	Breast Cancer				Diabetes				Iris				Wine			
	Best		Mean		Best		Mean		Best		Mean		Best		Mean	
Sim	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst	Trn	Tst
1	10.3	15.2	10.3	15.2	38.7	40.6	38.7	40.6	4.4	10.5	4.4	10.5	39.8	44.4	39.8	44.4
0.95	10.3	15.2	10.3	15.2	38.7	40.6	38.7	40.6	2.6	10.5	4.0	10.5	39.8	44.4	39.8	44.4
0.9	10.3	15.2	10.3	15.2	38.7	40.6	38.7	40.6	2.6	7.8	2.7	8.4	39.8	44.4	39.8	44.4
0.85	9.3	12.8	10.1	14.6	30.5	36.4	37.8	40.1	1.7	7.8	1.7	7.8	39.8	44.4	39.8	44.4
0.8	8	12.2	9.5	13.8	22.3	24.4	28.6	32.7	0.8	5.2	1.5	6.8	22.5	22.2	35.4	40.6
0.75	7	12.2	7.6	13.3	18.9	23.9	25.8	30.0	0	2.6	1.2	4.1	6.7	15.5	14.9	22.4
0.7	3.5	10.5	4.1	10.1	16.8	30.7	20.2	26.9	0	2.6	1.3	3.3	3.0	8.8	6.9	12.4
0.65	2.7	8.7	3.5	9.3	15.7	31.7	18.0	26.1	0	2.6	1.2	3.2	1.5	6.6	3.8	9.3
0.5	2.7	5.8	2.9	8.5	15.9	22.9	16.7	26.0	0	2.6	1.1	3.2	1.5	4.4	2.6	7.6

a number of classification problems. Empirically, the resulting model accuracy of microtuning with a similarity value larger than 0.6 is statistically undistinguishable from those obtained by free, unconstrained tuning.

It has been noted that the similarity value at which significant changes in model accuracy varies slightly against different problems investigated. It would be interesting to further examine whether such slight differences are mainly caused by the different number of rules generated or by the different cardinality of the input space. It would also be useful to determine how sensitive the proposed work would be to the similarity metric employed. Finally, Lozowski's algorithm is not the most accurate pure descriptive induction algorithm. It may be interesting to see how microtuning behaves when applied to optimised descriptive rules that use hedges [2], [3]. These tasks remain as important future work.

Acknowledgements

The second author is partly supported by the Fundación Marín-Blázquez, Spain. Whilst taking full responsibility for the views expressed here, both authors are grateful to Alexios Chouchoulas, Peter Ross and Antonio F. Gómez Skarmeta, for helpful discussions and assistance in the research reported.

References

[1] A. Lozowski, T. J. Cholewo, and J. M. Zurada. Crisp rule extraction from perceptron network classifiers. In *Proceedings of International Conference on Neural Networks*, volume Plenary, Panel and Special Sessions, pages 94–99, Washington, D.C., 1996.

[2] J. G. Marín-Blázquez and Q. Shen. From approximative to descriptive fuzzy classifiers. *IEEE Transactions on Fuzzy Systems*, to appear.

[3] J. Gómez Marín-Blázquez and Q. Shen. Linguistic hedges on trapezoidal fuzzy sets: A revisit. In *Proceedings of the 10th IEEE International Conference on Fuzzy Systems*, December 2001.

[4] J. Gómez Marín-Blázquez, Q. Shen, and A. Tuson. Tunning fuzzy membership functions with neighbourhood search techniques: A comparative study. In *Proceedings of the 3rd IEEE International Conference on Intelligent Engineering Systems*, pages 337–342, November 1999.

[5] M. Setnes, R. Babuska, and H. B. Verbruggen. Rule-based modeling: Precision and transparency. *IEEE Transactions on Systems, Man and Cybernetics - Part c: Applications and Reviews*, 28(1):165–169, Feb. 1998.

[6] Q. Shen and A. Chouchoulas. A modular approach to generating fuzzy rules with reduced attributes for the monitoring of complex systems. *Engineering Applications of Artificial Intelligence*, 13(3):263–278, 2000.

[7] R. Storn. On the usage of differential evolution for function optimization. In M. H. Smith, M. A. Lee, J. Keller, and J. Yen, editors, *1996 Biennial Conference of the North American Fuzzy Information Processing Society - NAFIPS*, pages 519–523. IEEE Service Center, June 19-22 1996.

[8] R. Storn and K. Price. Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Global Optimization*, 11:314–359, 1997.

[9] Uci machine learning databases. Available on web: `http://ftp.ics.uci.edu/pub/machine-learning-databases/`.

[10] J. Valente de Oliveira. Semantic constrains for membership function optimization. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, 29(1):128–138, Jan. 1999.

[11] L. X. Wang and J. M. Mendel. Generating fuzzy rules by learning from examples. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(6):1414–1427, November-December 1992.

[12] Lofti A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning i. *Information Sciences*, 8:199–249, 1975.

[13] Lofti A. Zadeh. Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems*, 4(2):103–111, May 1996.